

VIRTUAL BEACH FAQs

TABLE OF CONTENTS

[PREPARING TO USE VIRTUAL BEACH](#)

[OPERATING VIRTUAL BEACH](#)

[IMPORTING AND MANAGING DATA](#)

[MODEL BUILDING, CALIBRATION, AND VALIDATION](#)

[STATISTICAL MODELS AND APPROACHES](#)

[PREPARING TO USE VIRTUAL BEACH](#)

Q: What types of beaches are best statistically modeled?

A: Statistical models are best suited for beaches with an intermediate level of contamination, as models generally have trouble fitting rare events. The type of regression models that Virtual Beach produces will be best applied at beaches with FIB exceedance rates between 20-75%.

Q: How much historical data should be used to fit a model?

A: In general, we recommend using all available data to fit the model, unless the data are non-stationarity (have a temporal trend over the years). The mantra is “the more data, the better”. At the low end, 30-50 unique observations (rows of data) over 2 years may be sufficient. The key is capturing the most variability possible in the data (rainy days, sunny days, days with high E. coli, days with low E. coli, etc.).

Q: Where can I download Virtual Beach?

A: You can download Virtual Beach from the EPA [website](#) . A User's Guide written by Cyterski et al. is available [here](#).

As noted in the User's Guide: "Disk space requirements are about 140 MB for VB₃ and 170 MB for the DotNet Framework 4. The VB₃ application installer will attempt to download and install the DotNet Framework 4.0 if it is not already installed on the target system; this also requires a network connection. If necessary, a user can obtain the DotNet Framework 4 installer at no cost [here](#)."

OPERATING VIRTUAL BEACH

Q: I recently downloaded the newest version of VB and now I'm encountering errors when I run it. What should I do?

A: Installing a new version of VB in the same directory as an older version can cause odd errors. If possible, remove older versions before installing newer versions, and/or install any new version into its own folder.

Q: The Virtual Beach file I have only shows the prediction tab; none of the modeling tabs or associated data are visible. How do I fix this?

A: There are two types of Virtual Beach files – model (*.vb3m) and project (*.vb3p). The data and model-building tabs are hidden in model files so that these files can be shared with stakeholders without the risk of the model being accidentally changed or corrupted. This file type is not commonly used anymore. Now, most people save as project files, which show all the steps and available data in Virtual Beach. To convert a model file to project file, simply navigate to its folder and rename the file extension as .vb3p (i.e. change the 'm' to a 'p' in the filename).

Q: How do I fix the "out of memory" errors when operating VB (especially associated with the GBM and PLS models)?

A: For those systems that experience memory issues, look at this guidance:

<https://support.microsoft.com/en-us/help/15055/windows-7-optimize-windows-better-performance>

In particular, check your systems' swap space (virtual memory) configurations and adjust as recommended (near bottom of the page).

In addition, the most recent release of VB (3.0.6) made some changes to file management (saving and re-opening project files) because of memory issues/errors associated with these tasks, especially in regards to GBM and PLS modeling.

Q: Will I get the same results if I run the same model in different versions of VB?

A: Not always. Various versions of VB output different exceedance probabilities for the same model. Over the course of VB development, we have refined the exceedance probability calculations, so earlier versions likely show differences to later versions, both for VB 2.X and 3.X.

IMPORTING AND MANAGING DATA

Q: What should I do if I encounter an error when attempting to import data into Virtual Beach?

A:Below is a series of trouble-shooting options:

- The data file may be saved on a network drive, rather than local C drive, and the user does not have permission to access network drive. Try copying the data file to your local drive and try again.
- Check that the headers for your data do not include any spaces or special characters (e.g. “ “, %, # etc.).
- Make sure that the file you are trying to import is not open in another program. Virtual Beach cannot successfully import the data unless the original file is closed in all other programs.
- Check that the file is saved as an excel file (.xls). Excel 2016 files have known compatibility issues. The simplest solution is to save your data as an older excel file (e.g. 2010). Another option is to run the excel access database engine from Microsoft in order to properly translate the newer excel files (.xlsx) to Virtual Beach. Get (and then install) the 32-bit database engine from here: <https://www.microsoft.com/en-us/download/details.aspx?id=13255> the name of the correct installer is “AccessDatabaseEngine.exe” not “AccessDatabaseEngine_X64.exe” because VB is a 32-bit program, not 64-bit. Once you install this engine, your issue may clear your issue up.

- Check for “ghost data” in your spreadsheet. VB sometimes reads empty rows in your excel file as active when there is nothing visibly there. This could be from someone copying & pasting data into the spreadsheet and then deleting some rows or resizing the data rows. The best solution is to copy and paste the data you need into a fresh spreadsheet and then saving it as a completely different file.

Q: What do I do if I receive the following error when importing data in Virtual Beach?

"File read error: The 'Microsoft.Ace.OLEDB.12.0' provider is not registered on the local machine."

A: download and run the Microsoft Access Database Engine, which can be obtained here:
<http://www.microsoft.com/en-us/download/details.aspx?id=13255>

Q: If my data set for an independent variable (e.g. rainfall) has only a few non-zero values, should I still use it?

A: An independent variable that is mostly zeroes is typically uninformative. For any dataset less than 50 observations, you should have at least 20% non-zero values for any continuous independent variable. For datasets with 50 or more observations, you should have at least 10% non-zero observations, or at least 10 non-zero observations, whichever is the larger quantity. Using this rule, you would want at least 5 non-zero values for every IV given the smallest dataset that we recommend modeling (about 25 observations).

Q: How do I construct a model for predicting a variable with a temporal trend? (For example, FIB levels at the beach are slowly rising/ falling over the years – i.e. “non-stationary”)

A: If developing a model for prediction of water quality for the current swim season, limit the number of previous years’ data used to develop the model, as data further back in time may be much less relevant given the temporal changes at the beach. There are various reasons that FIB levels may exhibit non-stationarity, especially in urban/developing areas or due to continued beach restoration activity. Addressing stationarity is beyond the current intention of the software. A more technical user can address this problem using more advanced statistical software (R/SAS, etc.)

MODEL BUILDING, CALIBRATION, AND VALIDATION

Q: What are the most common variables and variable transformations used to build a model?

A: General Variables

Air Temperature

Water Temperature

Cloud Cover

Beach Water Turbidity

Rainfall (24 hr total)

Rainfall (48 hr total)

Rainfall (72 hr total)

Vector Variables

The following variables – wind, current, and wave – are converted into offshore and alongshore components by Virtual Beach using the beach angle. These transformed variables (e.g. “WindA_comp” and “WindO_comp”) are used by the model.

Variable	Notes	Data Source
Wind speed/ velocity (WSPD)		Great Lakes Coastal Forecasting System, NOAA
Wind direction (WDIR)	0 - 359 degrees, where 0 is out of the North	Great Lakes Coastal Forecasting System, NOAA
Current speed/ velocity (CSPD)	Offshore wave height	Great Lakes Coastal Forecasting System, NOAA
Current Direction (CDIR)	0 - 359 degrees, where 0 is toward the North	Great Lakes Coastal Forecasting System, NOAA
Wave height (WVHT)		Great Lakes Coastal Forecasting System, NOAA
Wave direction (WVDIR)	0 - 360 degrees, where 0 is toward the North	Great Lakes Coastal Forecasting System, NOAA

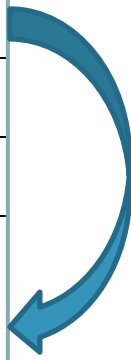
Variable Combinations

Since Virtual Beach requires all variables to be numerical/ quantitative, many qualitative variables are converted to numbers by using a “binary” yes/ no approach. For example, measurements of water clarity are often qualitatively categorized as clear, slightly turbid, turbid etc. To convert this to numerical values for use in VB, a “binary” approach could be used where 1 = condition in that category existed and 0 = condition in that category did *not* exist (see table below). To combine similar categories, you can add two categories.

Example of combining two similar data columns (Turbid + Opaque)

Turbidity	Clear	Slightly Turbid	Turbid	Opaque
Day 1	0	0	1	0
Day 2	0	1	0	0
Day 3	0	0	0	1

Turbidity	Clear	Slightly Turbid	Turbid
Day 1	0	0	1
Day 2	0	1	0
Day 3	0	0	1



Variable	Example VB formula	Notes
Gulls*Wave height	PROD[GULLS,WAVEHEIGHT_FT]	This combination addresses an issue with collinearity between two potentially important variables for predicting E. Coli. Wave height normally influences the number of gulls present (higher waves, fewer gulls riding it out). But, both wave height and gulls can directly influence E. Coli. Strong collinearity is more of a concern
Turbid + Opaque	SUM[TURBID, OPAQUE]	This combination reduces the number of variables considered and combines two categories that make have similar meaning.

The following variable combinations are to get a measure of the “flashiness” of the system

Variables	Example VB formula
= 24 Hour Maximum – 24 Hour Mean of Tributary Discharge	DIFF[TRIBmax24,TRIB24]
=24 Hour Maximum – 24 Hour Minimum of Tributary Discharge	DIFF[TRIBmax24,TRIBmin24]
= 24 Hour Maximum– 6 Hour Mean of Tributary Discharge	DIFF[TRIBmax24,TRIB6]
= 48 Hour Maximum– 6 Hour Mean of Tributary Discharge	DIFF[TRIBmax48,TRIB6]
= 48 Hour Maximum–24 Hour Maximum of Tributary Discharge	DIFF[TRIBmax48,TRIBmax24]
= 48 Hour Maximum–48 Hour Mean of Tributary Discharge	DIFF[TRIBmax48,TRIB48]
= 48 Hour Maximum–24 Hour Minimum of Tributary Discharge	DIFF[TRIBmax48,TRIBmin48]

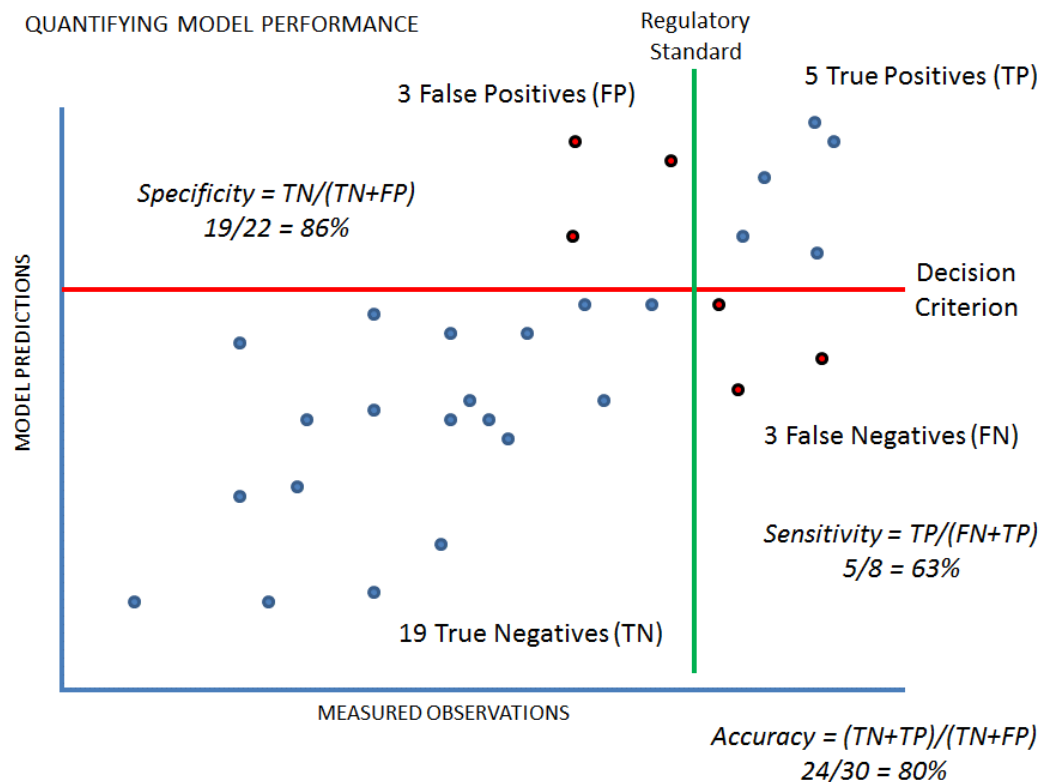
Q: How do I interpret the results for model specificity, sensitivity, and accuracy? How are these used to indicate a “good” model?

A: See plot below for an example calculation of these quantities. If we plot model predictions vs. actual observations, and specify a model decision criterion (DC) and regulatory standard (RS), we create four quadrants on the plot:

Specificity = True Negatives / (True Negatives + False Positives)

Sensitivity = True Positives / (True Positives + False Negatives)

Accuracy = (True Positives + True Negatives)/ Total Predictions



When developing a model, aim for a specificity of at least 80% and a sensitivity of at least 50%.

Q: If I have too much time on my hands, how can I manually calculate the probability of exceedance for a prediction?

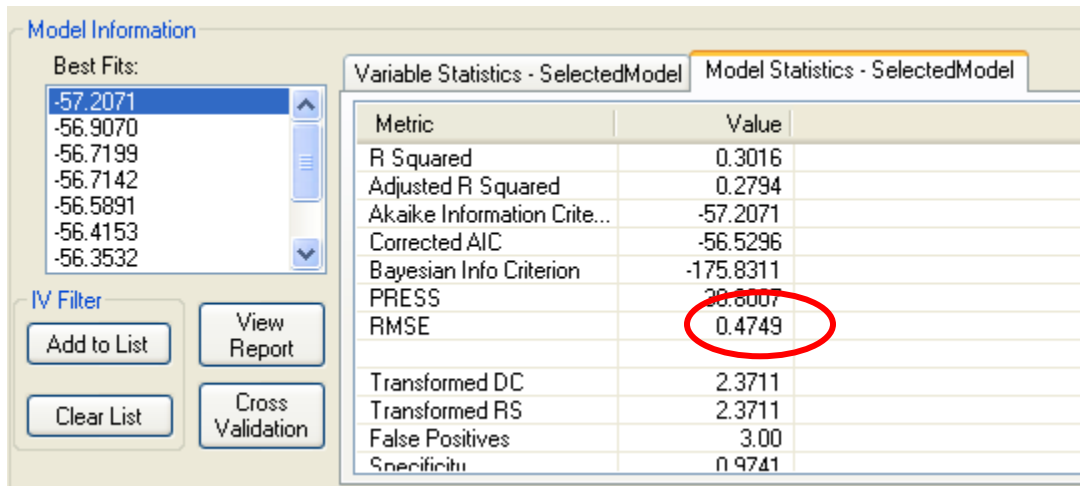
A: Matrix algebra is needed to manually calculate the probability of exceedance for a given prediction. Say we have this conceptual regression model:

$$\text{Ecoli Concentration} = \text{Intercept} + a \cdot \text{Water Temp} + b \cdot \text{Rainfall} + c \cdot \text{Turbidity}$$

To calculate the probability of exceedance for any prediction, we'll first need to calculate the standard error of that prediction:

$$\text{Standard Error of Prediction (SEP)} = \text{RMSE} * (1 + x'(X'X)^{-1}x)^{0.5}$$

Where RMSE is the “root mean square error” of the fitted model. VB gives you this quantity on the “Model Statistics” tab:



Metric	Value
R Squared	0.3016
Adjusted R Squared	0.2794
Akaike Information Crite...	-57.2071
Corrected AIC	-56.5296
Bayesian Info Criterion	-175.8311
PRESS	28.6007
RMSE	0.4749
Transformed DC	2.3711
Transformed RS	2.3711
False Positives	3.00
Specificity	0.9741

Next, you'll need the vector of independent variable values for the prediction you want to make – this is small “x” in the equation. For example, if you are making a prediction when Water Temp is 65, Rainfall is 1.7 and Turbidity is 75 NTU, then “x” would be (1, 65, 1.7, 75). The initial “1” represents the model intercept.

The final thing needed is capital “X,” which is the matrix of the original values of Temp, Rainfall, and Turbidity that the model was constructed with. So say we built the model using 30 observations. Big “X” could look like this:

Intercept	Water Temp	Rainfall	Turbidity
1	74.4	2.1	124.6
1	74.8	0.6	93.1
1	73.6	0.0	49.5
1	69.8	0.0	85.4
1	67.6	1.5	90.9
1	71.7	0.0	82.8
1	68.2	0.0	63.3
1	69.8	1.0	133.4
1	67.0	1.8	106.4
1	69.8	0.0	99.3
1	69.4	0.0	62.0
1	74.9	0.0	127.4
1	65.2	0.0	67.5
1	66.2	0.0	129.9
1	70.1	0.4	78.6
1	69.3	0.0	51.1
1	70.9	0.0	121.1
1	73.7	0.0	48.5
1	69.9	1.1	101.5
1	66.7	0.5	35.8
1	71.9	1.5	44.9
1	69.8	0.0	35.7
1	74.3	0.0	130.0
1	70.6	0.0	94.3
1	74.1	0.0	63.1
1	72.6	0.0	70.8
1	70.0	0.0	63.0
1	65.3	0.4	85.9
1	71.8	0.0	59.2
1	74.0	1.3	124.0

In the equation above for the SEP, X' is the transpose of X , and x' is the transpose of x . It may be easiest to first calculate $X'X$, then take the inverse of that product (which can be done in Excel using the function "minverse"), then multiplying this result by x' , then multiply that result by x (which should return a single scalar quantity), then add 1, and then finally take the square root to get the SEP.

Note that if you have two IVs, $X'X$ is a 3x3 matrix; if you had three predictors, $X'X$ would be a 4x4 matrix; if you had four predictors, $X'X$ would be a 5x5 matrix, etc. (the extra dimension is because of the model intercept).

Remember, if " x " is a row vector like (1,2,4,3,5,3,2,4), then x' is a column vector:

- 1
- 2
- 4
- 3

5
3
2
4

If the matrix "X" is:

1	3	7	6
1	4	5	5
1	7	5	2
1	5	9	6
1	2	3	9
1	3	3	7
1	1	7	4
1	9	8	5
1	5	7	6
1	6	7	6
1	3	4	2
1	9	8	6
1	5	7	3
1	7	7	2
1	1	1	8

Then the matrix X' is:

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

3 4 7 5 2 3 1 9 5 6 3 9 5 7 1

7 5 5 9 3 3 7 8 7 7 4 8 7 7 1

6 5 2 6 9 7 4 5 6 6 2 6 3 2 8

Once we have the SEP calculated, we need two other things:

- 1) the predicted value from the model
- 2) the decision criterion or threshold that you want to determine the probability of exceeding

Once you have these three (SEP, predicted value, decision criterion), you can plug them into a normal distribution probability calculator. The excel function "normdist" is one such calculator, but most statistical packages should have one.

= 100*NORMDIST(PRED, DC, SEP, "TRUE")

The "TRUE" designation means you want a cumulative probability from the function.

So, back to our example where Water Temp = 65, Rainfall = 1.7, and Turbidity =75. Given, this input, the model prediction is 217, the calculated SEP is 110.7, and the decision criterion is 235. In Excel, type in:

= 100*NORMDIST(217, 235, 100.7, TRUE) = 42.9

So we have a 42.9% chance of an exceedance (>235) for this predicted value. If the model prediction was 247, and we had the same SEP:

= 100*NORMDIST(247, 235, 100.7, TRUE) = 54.7% of having an exceedance.

Note the slight difference when calculating a probability of exceedance for a fitted model value versus the probability of exceedance for a new prediction:

$$\text{Standard Error of Fit Value (SEF)} = \text{RMSE} * (x'(X'X)^{-1}x)^{0.5}$$

$$\text{Standard Error of Prediction (SEP)} = \text{RMSE} * (1 + x'(X'X)^{-1}x)^{0.5}$$

You can see that the SEP is a little bit larger than the SEF value, because it's a new prediction made with data not included in the original model fitting process – that makes it a little less certain.

Q: What is the “Decision Criterion”?

A: The Decision Criterion (DC) is unique to the model and the result of the optimization process of adjusting to maximize the sensitivity and specificity values. It can be thought of as the as the regulatory standard (for example, 235 CFU) calibrated to the model. Put another way, it is a sort of calibration line used to convert predictions to actual observations. Adjusting the DC allows you to recognize the bias in the model and account for how Virtual Beach’s model predictions for E. coli tend to underestimate (are “muted”) compared to actual observed concentrations.

Q: How can I adjust the model “Decision Criterion” after fitting the model?

A: After fitting a model, the user is free to move the decision criterion (DC) up and down to see what DC maximizes/optimizes the model’s sensitivity and specificity. One should experiment with the DC to strike a happy balance between sensitivity and specificity, while incorporating value judgments about each (high specificity means you don’t often close the beach unnecessarily, while high sensitivity means you rarely expose swimmers to high FIB concentrations).

Most people seem to value sensitivity over specificity, but how much more? By lowering the DC, you may incur one additional false positive, but if you lose one false negative, that may be worthwhile. However, if you incur multiple false positives to delete one false negative by lowering the DC, that may not be palatable.

Some people like to pick the DC that maximizes the overall accuracy of the model. In the end, the DC is a cutoff that says, “If the model prediction is above this number, we will issue an advisory/close the beach, if the model prediction is below this number, we won’t.” It is completely up to the decision maker, while the Regulatory Standard is set by law or proclamation and should not be adjusted.

Setting the decision criterion to some value not equal to the regulatory standard troubles some users. However, when we adopt a modeling approach, we have decided to base our advisory decisions on a statistical model derived from data collected over multiple beach seasons. You should not think of the model predictions as actual FIB concentrations, but only some quantity that is related to actual concentrations. When you lower the DC, you are saying that a model prediction of 150, for example, is roughly equivalent to an actual FIB concentration of 200, for example. You are acknowledging some difference exists between the model predictions and the actual concentrations, but you are factoring that difference into the decision making process.

Q: How do I use the “variable influence” measures given in the GBM and PLS tabs for variable selection?

A: Some anecdotal experimentation has shown that better GBM/PLS results are obtained if only one variable at a time is removed (i.e., the variable with the lowest influence), then the model is re-run after each removal, rather than deleting all of the variables with seemingly low influence at once.

Q: Why do the GBM model results change from run to run?

A: This occurs because the GBM fit is determined across many iterative steps, and in each step a random subset of the data are chosen for residual determination. This stochasticity actually helps prevent overfitting (fitting the model too closely to the actual data, leading to poorer predictive performance). If you want reproducibility (getting the exact same model as you did before), on the “Model” subtab in GBM, check the “Set Seed Value” box (see picture below), and type in some number there. Then, the next time you run the model, check that box and put in the same number. This will give you the same model as you produced previously.

Virtual Beach 3

File Location Global Datasheet **GBM** MLR PLS

Compute A O
 Manipulate
 Transform
 Run
 Cancel
 Drop Variable(s)

Manipulate Data Model Variable Selection

Data Manipulation Variable Selection **Model** Diagnostics

Model Evaluation Threshold

235 Regulatory Standard

Threshold entry is transformed:

Value
 Log10 (value)
 Loge (value)
 Power (value) exp: 1

2013 US Regulatory Standards

E. coli, Freshwater: 235
 Enterococci, Freshwater: 61
 Enterococci, Saltwater: 104

Decision Criterion Set Seed Value: 3227

Model Summary

Variable	Coefficient	Influence

Model Validation

True Positives	True Negatives	False Positives	False Negatives	Sen

STATISTICAL MODELS AND APPROACHES

Q: Can you provide more details about what the GBM is, and how it operates?

The Generalized Boosted Regression Model (GBM, also known as a gradient boosting machine) uses binary decision rules (grouped together as a decision/regression “tree”) to arrive at predictions of a response variable. For example, one such rule might be “If turbidity ≥ 15 NTU, increase/decrease the expected FIB concentration by some amount.” The innovative aspect of GBM is that the algorithm doesn’t solve for a single, complex decision tree: it builds a hierarchical set of simple trees, with each subsequent tree fit to the residual error from the previous tree. GBM avoids overfitting by developing each new tree based on a random set of these residual values.

While each tree is a simple structure, the long, linear combination of regression trees is more complicated. A negative aspect of a GBM model is that the model cannot easily be inspected graphically or expressed mathematically – it is something of a “black box.” However, what it lacks in interpretability and transparency can often be made up in terms of prediction accuracy. Another noted aspect of GBM, unlike MLR and PLS, is that it handles non-linear relationships between the response and independent variables (IV’s) without the need for transformations.

GBM is best used on datasets with > 50 -100 observations; instability of the solution can occur on smaller datasets. The GBM method in Virtual Beach uses an algorithm for determining an optimal decision criterion for a fitted GBM model by striving for a balance between true negative and true positive outcomes. The GBM routine will not run if the dataset has no observations above the designated regulatory standard, so the user may be best served by defining the regulatory standard somewhere near the 75th percentile of the response variable distribution; this would provide a good number of observations on which to base the choice of a decision criterion.

Q: What is the thought process behind basing variable inclusion decisions on univariate correlations of independent variables with the response variable?

A: Independent variables can definitely emerge that explain variability in the response variable only when other variables are already present in the model. Say you have variable, X1, that has a decent linear relationship to the response variable, Y. You then look to see if any other X variables, like X2, have a relationship to the residuals of that Y vs X1 regression. You may find that X2 has a fair correlation with the residuals of Y vs X1, but if you just looked at the relationship of Y to X2, you would not see much. That is what is happening when certain variables emerge as important in the MLR model with several independent variables, even though their univariate correlation with the response is not significant.

Q: What is the general philosophy on parsimony in model fitting methodology, and emphasis on predictive, rather than fitting, performance of models?

Most statisticians would prefer a parsimonious model (as few IVs as possible), where all of the variables have p-values < 0.05 . The higher the p-value, the higher the likelihood that the relationship between Y and the IV is spurious. We recommend using the corrected AIC or the BIC as evaluation criterion, as these are the most stringent, and will lead to the most parsimonious models.

Model fit (as measured by R-squared) will always get larger as you add more terms, but the ability of the model to make accurate predictions often declines. Model developers will often look at cross-validation statistics to see how each model performs in predictive mode. Using the PRESS statistic as a model evaluation criterion puts emphasis on how the model behaves when predicting observations not used to fit the model.

When running the cross-validation routine from the MLR tab for any model, choose 5000-10000 trials in order to stabilize the root mean square error of prediction (RMSEP) that is output by that routine.

A user may find that a model's RMSEP declines if an IV with a fairly high p-value (≥ 0.15) is included. In this case, the user may want to retain that variable in the model, but traditionalists would argue otherwise.